CSE 258 – Lecture 8

Web Mining and Recommender Systems

Extensions of latent-factor models,

(and more on the Netflix prize!)

Extensions of latent-factor models

So far we have a model that looks like:

$$f(u,i) = \alpha + \beta_u + \beta_i + \gamma_u \cdot \gamma_i$$

How might we extend this to:

- Incorporate features about users and items
- Handle implicit feedback
- Change over time

See **Yehuda Koren** (+Bell & Volinsky)'s magazine article:
"Matrix Factorization Techniques for Recommender Systems"
IEEE Computer, 2009

Extensions of latent-factor models

1) Features about users and/or items

(simplest case) Suppose we have **binary attributes** to describe users or items

$$A(u) = [1,0,1,1,0,0,0,0,0,1,0,1]$$

attribute vector for user *u*

e.g. is female   is male   is between 18-24yo

Extensions of latent-factor models

1) Features about users and/or items

(simplest case) Suppose we have **binary attributes** to describe users or items

- Associate a **parameter vector** with each attribute
- Each vector encodes how much a particular feature "offsets" the given latent dimensions

$$A(u) = [1,0,1,1,0,0,0,0,0,1,0,1]$$

attribute vector for user *u*

e.g. y_0 = [-0.2,0.3,0.1,-0.4,0.8]
~ "how does being male impact gamma_u"

Extensions of latent-factor models

1) Features about users and/or items

(simplest case) Suppose we have **binary attributes** to describe users or items

- Associate a **parameter vector** with each attribute
- Each vector encodes how much a particular feature "offsets" the given latent dimensions
- Model looks like:

$$f(u,i) = \alpha + \beta_u + \beta_i + (\gamma_u + \textstyle\sum_{a \in A(u)} \rho_a) \cdot \gamma_i$$

- Fit as usual:

$$\arg\min_{\alpha,\beta,\gamma,\rho} \textstyle\sum_{u,i \in \text{train}} \underbrace{(f(u,i) - r_{u,i})^2}_{\text{error}} + \underbrace{\lambda\Omega(\beta,\gamma)}_{\text{regularizer}}$$

Extensions of latent-factor models

2) Implicit feedback

Perhaps many users will never actually rate things, but may still interact with the system, e.g. through the movies they view, or the products they purchase (but never rate)

- Adopt a similar approach – introduce a binary vector describing a user's actions

$$N(u) = [1,0,0,0,1,0,....,0,1]$$

implicit feedback vector for user *u*

e.g. y_0 = [-0.1,0.2,0.3,-0.1,0.5]
Clicked on "Love Actually" but didn't watch

Extensions of latent-factor models

2) Implicit feedback

Perhaps many users will never actually rate things, but may still interact with the system, e.g. through the movies they view, or the products they purchase (but never rate)

- Adopt a similar approach – introduce a binary vector describing a user's actions
  - Model looks like:

$$f(u,i) = \alpha + \beta_u + \beta_i + (\gamma_u + \frac{1}{\|N(u)\|} \sum_{a \in N(u)} \rho_a) \cdot \gamma_i$$

normalize by the number of actions the user performed

Extensions of latent-factor models

3) Change over time

There are a number of reasons why rating data might be subject to temporal effects…

Extensions of latent-factor models

3) Change over time

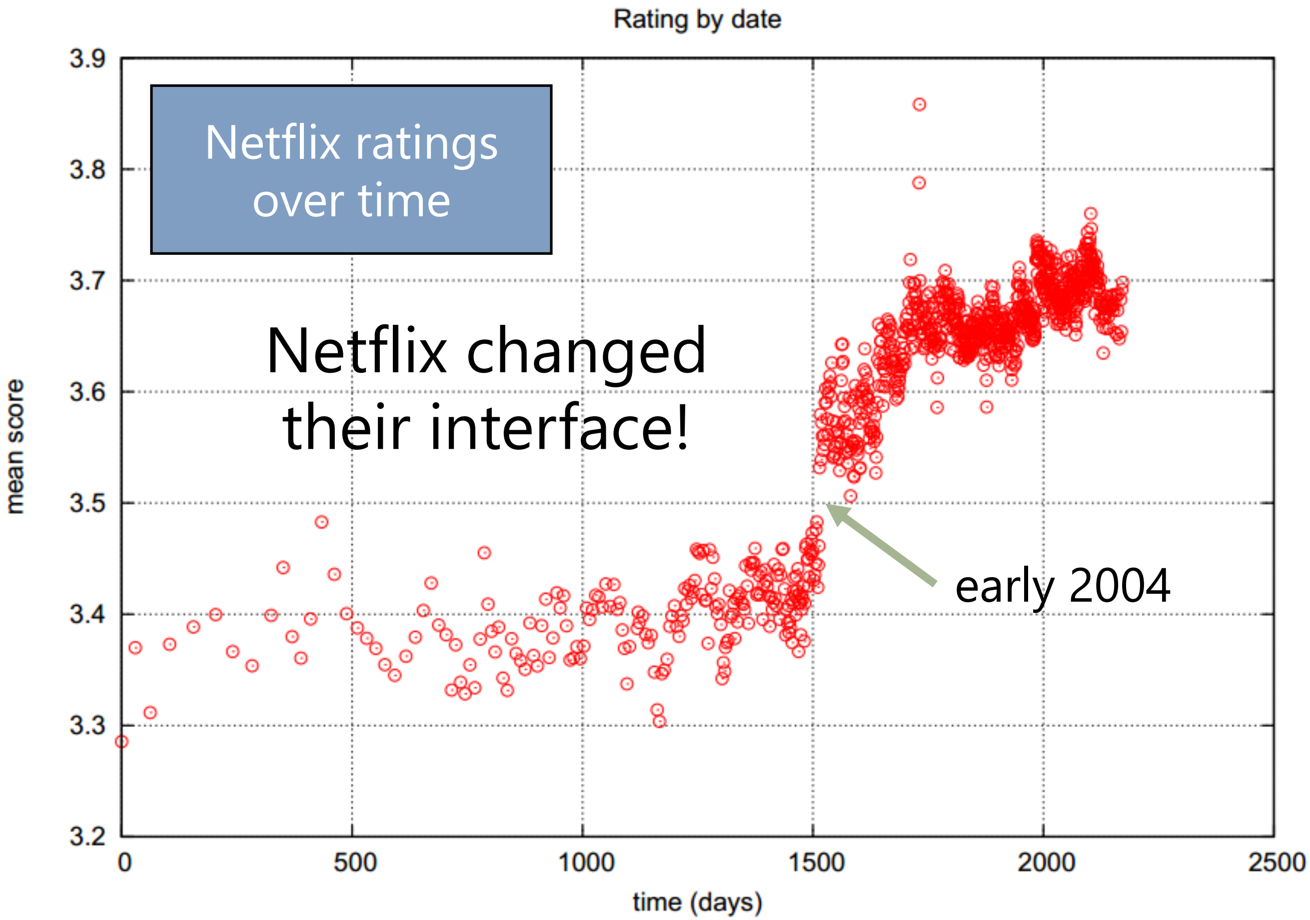
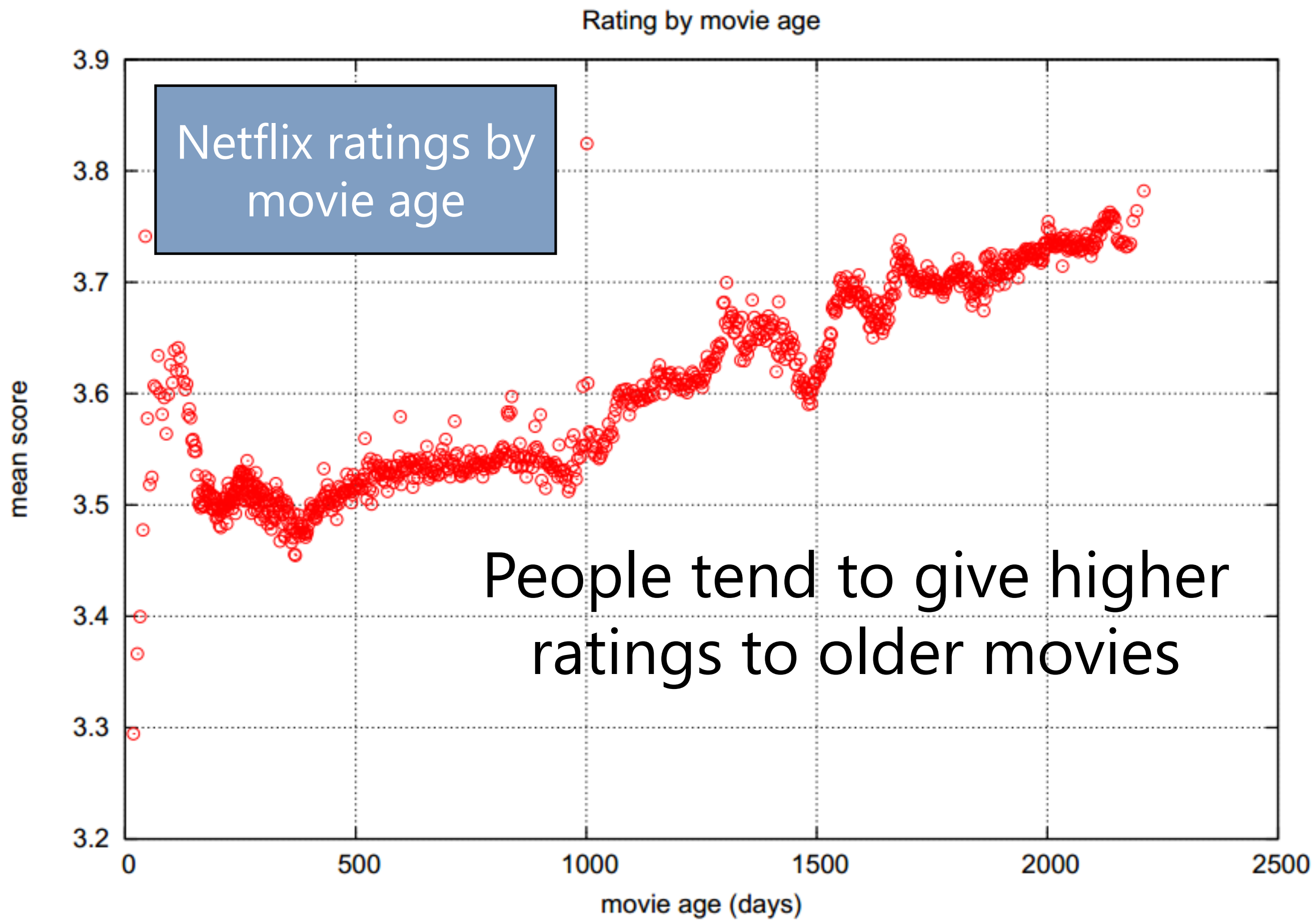


Figure from Koren: "Collaborative Filtering with Temporal Dynamics" (KDD 2009)

Extensions of latent-factor models

3) Change over time



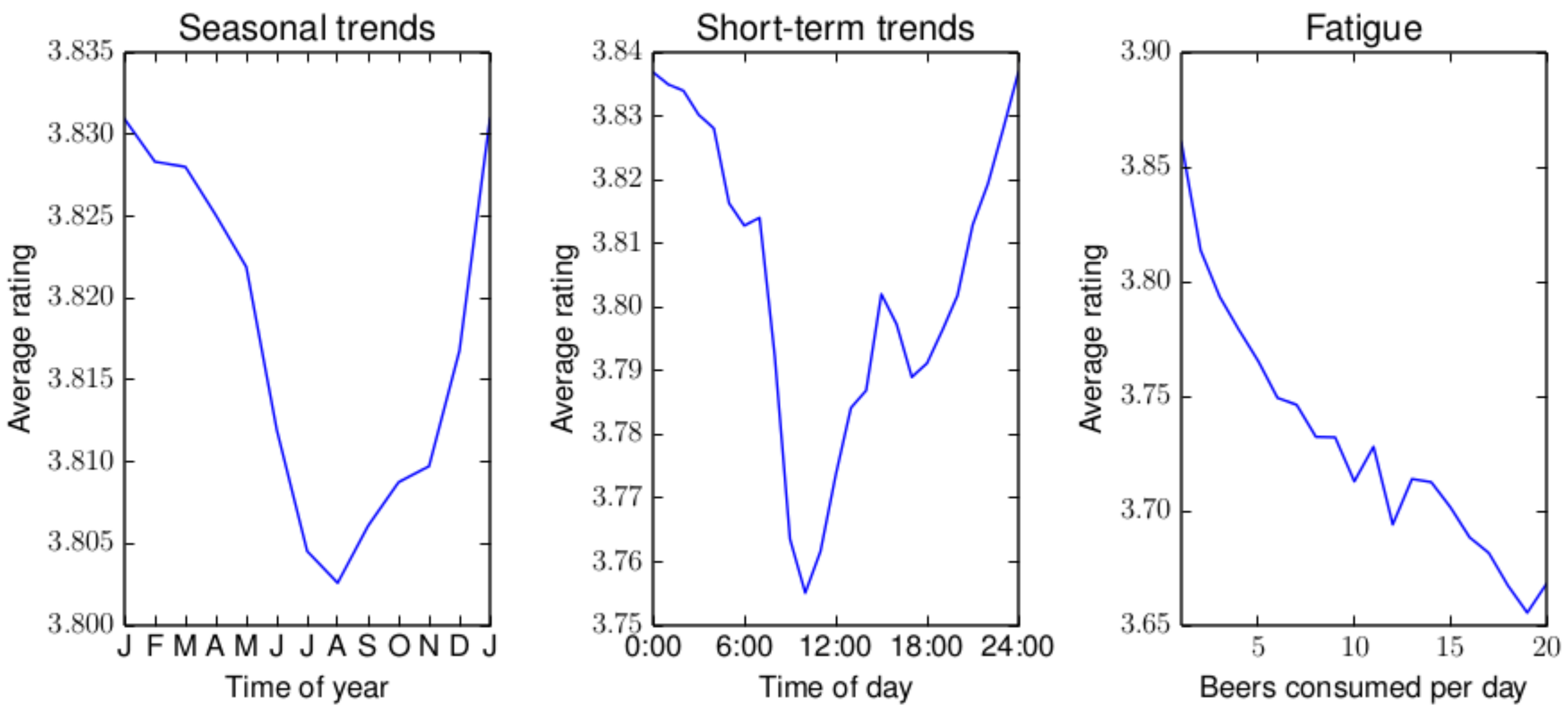
Netflix ratings by movie age

People tend to give higher ratings to older movies

Figure from Koren: “Collaborative Filtering with Temporal Dynamics” (KDD 2009)

Extensions of latent-factor models

3) Change over time



A few temporal effects from beer reviews

Extensions of latent-factor models

3) Change over time

There are a number of reasons why rating data might be subject to temporal effects…

e.g. "Collaborative filtering with temporal dynamics" Koren, 2009

e.g. "Sequential & temporal dynamics of online opinion" Godes & Silva, 2012

e.g. "Temporal recommendation on graphs via long- and short-term preference fusion" Xiang et al., 2010

e.g. "Modeling the evolution of user expertise through online reviews" McAuley & Leskovec, 2013

- Changes in the interface
- People give higher ratings to older movies (or, people who watch older movies are a biased sample)
- The community's preferences gradually change over time
- My girlfriend starts using my Netflix account one day
- I binge watch all 144 episodes of buffy one week and then revert to my normal behavior
- I become a "connoisseur" of a certain type of movie
- Anchoring, public perception, seasonal effects, etc.

Extensions of latent-factor models

3) Change over time

Each definition of temporal evolution demands a slightly different model assumption (we'll see some in more detail later tonight!) but the basic idea is the following:

1) Start with our original model:

$$f(u,i) = \alpha + \beta_u + \beta_i + \gamma_u \cdot \gamma_i$$

2) And define some of the parameters as a function of time:

$$f(u,i,t) = \alpha + \beta_u(t) + \beta_i(t) + \gamma_u(t) \cdot \gamma_i$$

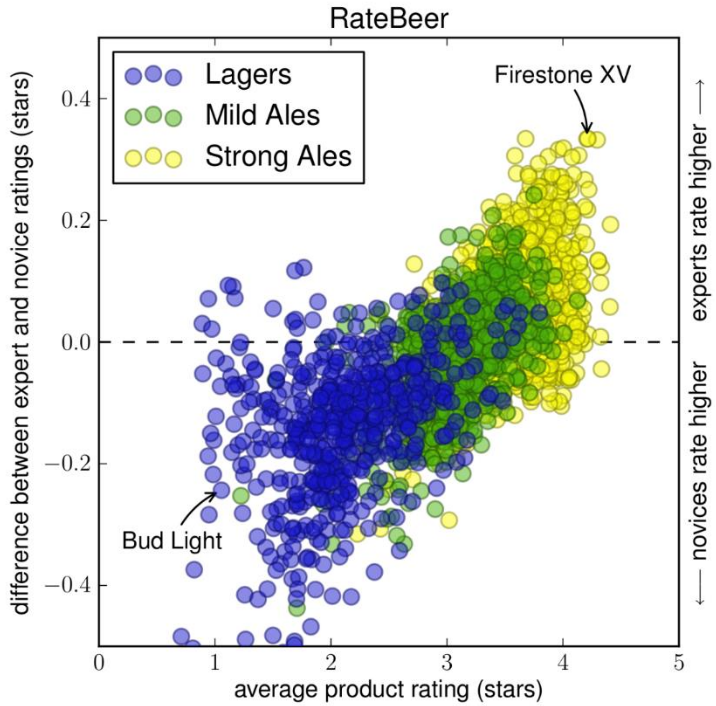
3) Add a regularizer to constrain the time-varying terms:

$$\arg\min_{\alpha,\beta,\gamma} \sum_{u,i,t \in \mathrm{train}} (f(u,i,t) - r_{u,i,t})^2 + \lambda_1 \Omega(\beta,\gamma) + \underbrace{\lambda_2 \|\gamma(t) - \gamma(t+\delta)\|}_{\text{parameters should change smoothly}}$$

Extensions of latent-factor models

3) Change over time

Case study: how do people acquire tastes for beers (and potentially for other things) over time?



Differences between "beginner" and "expert" preferences for different beer styles

Extensions of latent-factor models

4) Missing-not-at-random

- Our decision about whether to purchase a movie (or item etc.) is a function of how we **expect** to rate it
- Even for items we've purchased, our decision to **enter a rating** or write a review **is a function of our rating**
  - e.g. some rating distribution from a few datasets:

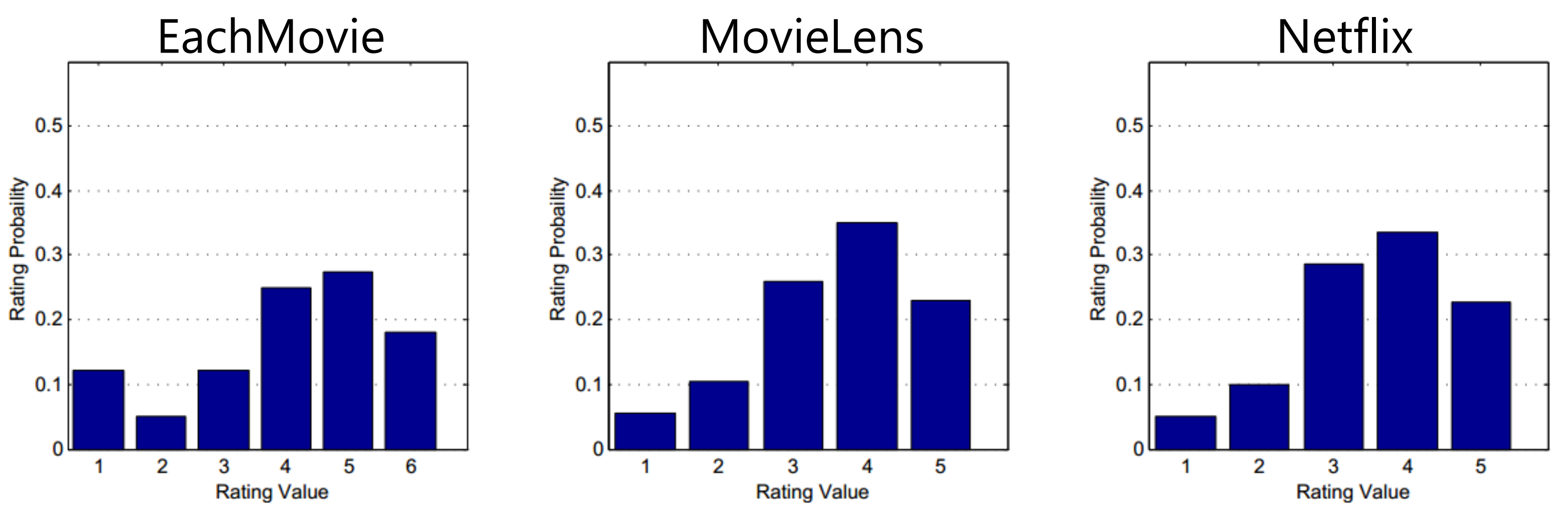
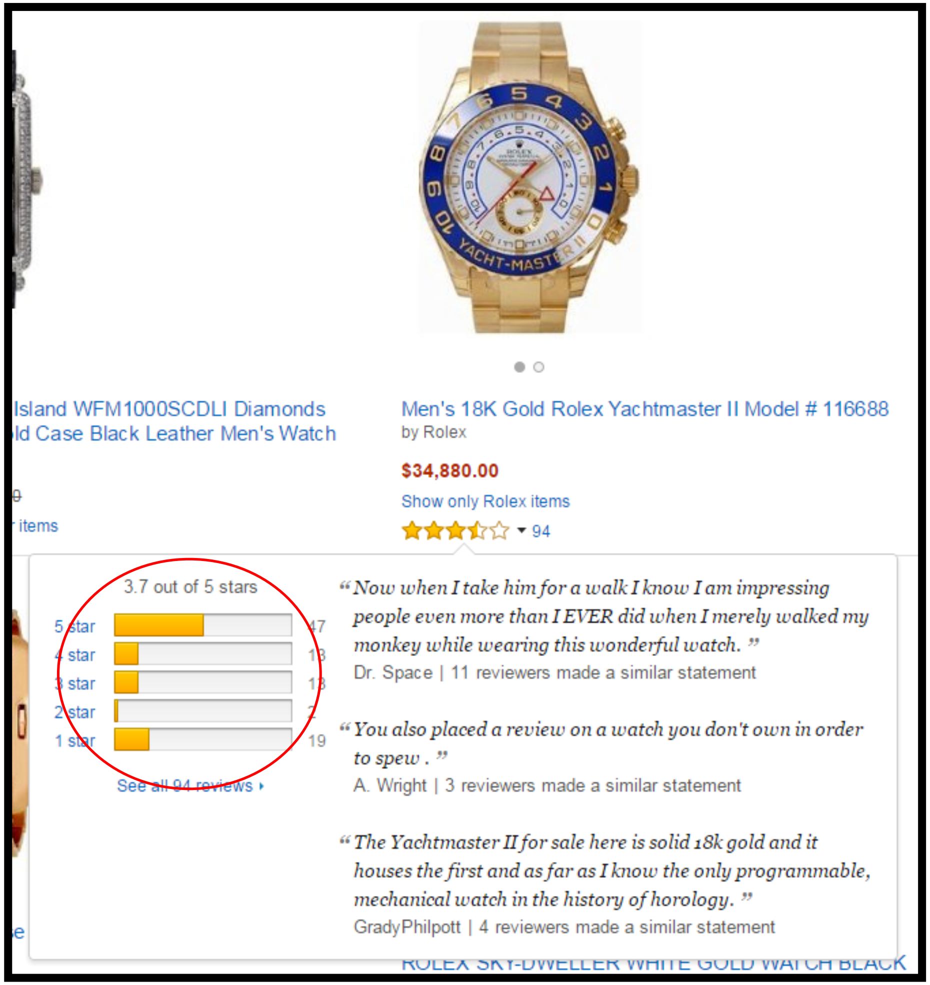


Figure from Marlin et al. "Collaborative Filtering and the Missing at Random Assumption" (UAI 2007)

Extensions of latent-factor models

4) Missing-not-at-random

e.g. Men's watches:



Extensions of latent-factor models

4) Missing-not-at-random

- Our decision about whether to purchase a movie (or item etc.) is a function of how we **expect** to rate it
- Even for items we've purchased, our decision to **enter a rating** or write a review **is a function of our rating**
- So we can predict ratings more accurately by building models that account for these differences

1. Not-purchased items have a different prior on ratings than purchased ones
2. Purchased-but-not-rated items have a different prior on ratings than rated ones

Figure from Marlin et al. "Collaborative Filtering and the Missing at Random Assumption" (UAI 2007)

Moral(s) of the story

How much do these extension help?

Moral: increasing complexity helps a bit, but changing the model can help **a lot**

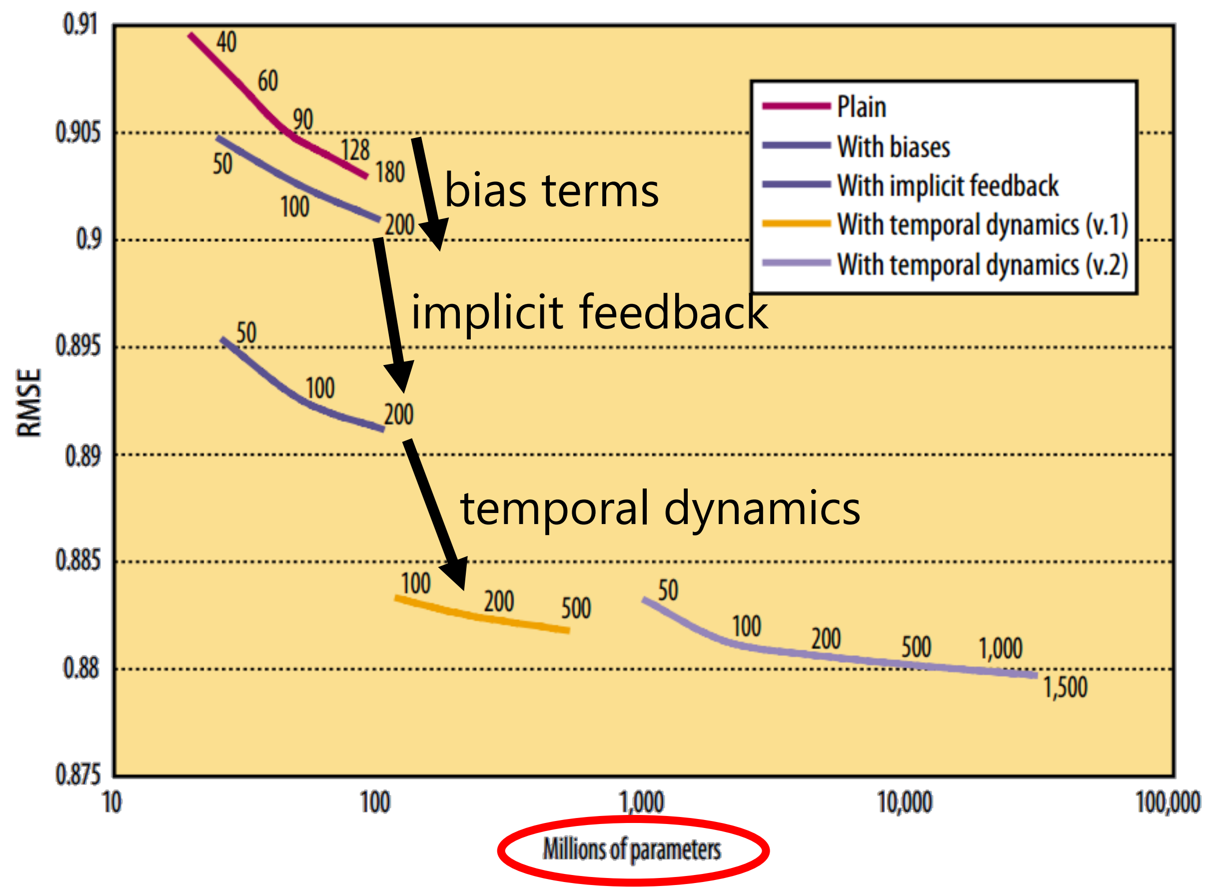


Figure from Koren: “Collaborative Filtering with Temporal Dynamics” (KDD 2009)

Moral(s) of the story

So what actually happened with Netflix?

- The AT&T team "BellKor", consisting of Yehuda Koren, Robert Bell, and Chris Volinsky were early leaders. Their main insight was how to effectively incorporate temporal dynamics into recommendation on Netflix.
- Before long, it was clear that no one team would build the winning solution, and Frankenstein efforts started to merge. Two frontrunners emerged, "BellKor's Pragmatic Chaos", and "The Ensemble".
- The BellKor team was the first to achieve a 10% improvement in RMSE, putting the competition in "last call" mode. The winner would be decided after 30 days.
- After 30 days, performance was evaluated on the hidden part of the test set.
- Both of the frontrunning teams had **the same** RMSE (up to some precision) but BellKor's team submitted their solution 20 minutes earlier and won $1,000,000

For a less rough summary, see the Wikipedia page about the Netflix prize, and the nytimes article about the competition: http://goo.gl/WNpy7o

Moral(s) of the story

Afterword

- Netflix had a class-action lawsuit filed against them after somebody de-anonymized the competition data
- $1,000,000 seems to be **incredibly cheap** for a company the size of Netflix in terms of the amount of research that was devoted to the task, and the potential benefit to Netflix of having their recommendation algorithm improved by 10%
- Other similar competitions have emerged, such as the Heritage Health Prize ($3,000,000 to predict the length of future hospital visits)

- But… the winning solution never made it into production at Netflix – it's a monolithic algorithm that is very expensive to update as new data comes in*

*source: a friend of mine told me and I have no actual evidence of this claim

Moral(s) of the story

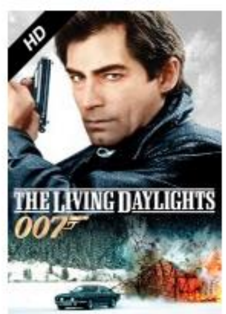
Finally...

Q: Is the RMSE really the right approach? Will improving rating prediction by 10% actually improve the user experience by a significant amount?
A: Not clear. Even a solution that only changes the RMSE slightly could drastically change which items are top-ranked and ultimately suggested to the user.
Q: But... are the following recommendations actually any good?
A1: Yes, these are my favorite movies!
or **A2:** No! There's no **diversity**, so how will I discover **new** content?

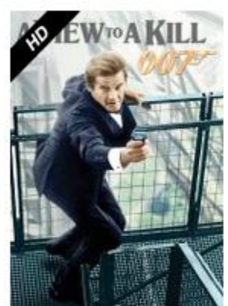
5.0 stars

5.0 stars

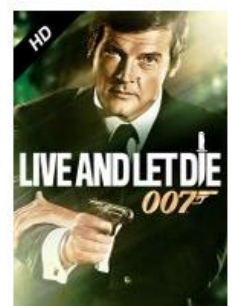
5.0 stars



5.0 stars



4.9 stars



4.9 stars

4.8 stars

4.8 stars

predicted rating

Summary

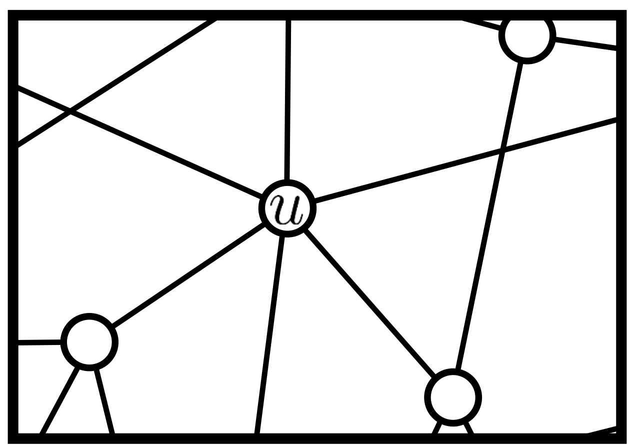
Various extensions of latent factor models:

- Incorporating features
e.g. for cold-start recommendation
- Implicit feedback
e.g. when ratings aren't available, but other actions are
- Incorporating temporal information into latent factor models
seasonal effects, short-term "bursts", long-term trends, etc.
- Missing-not-at-random
incorporating priors about items that were not bought or rated
- The Netflix prize

Things I didn't get to...

Socially regularized recommender systems

see e.g. "Recommender Systems with Social Regularization"
http://research.microsoft.com/en-us/um/people/denzho/papers/rsr.pdf



network

$$\arg\min_{\alpha,\beta,\gamma} \sum_{u,i\in\text{train}} (f(u,i) - r_{u,i})^2 + \lambda_1 \Omega(\beta,\gamma) + \underbrace{\lambda_2 \sum_{u,v\in\mathcal{E}} \|\gamma_u - \gamma_v\|}_{\text{social regularizer}}$$

Questions?

Further reading:

Yehuda Koren's, Robert Bell, and Chris Volinsky's IEEE computer article:
http://www2.research.att.com/~volinsky/papers/ieeecomputer.pdf
Paper about the "Missing-at-Random" assumption, and how to address it:
http://www.cs.toronto.edu/~marlin/research/papers/cfmar-uai2007.pdf
Collaborative filtering with temporal dynamics:
http://research.yahoo.com/files/kdd-fp074-koren.pdf
Recommender systems and sales diversity:
http://papers.ssrn.com/sol3/papers.cfm?abstract_id=955984